

Berichts.-Nr.:

DLR-IB 224-2014 A 105

Verfasser:

**Steffen Risius, Ulrich Henne
Werner Sachs**

Titel:

**Anleitungen für das
TSP-Messsystem am
European Transonic Windtunnel
(ETW)**

Datum: 15.08.2014

Auftraggeber:

Förderkennz.:

Vorgesehen für:

Der Bericht

31 Seiten einschl.

0 Tabellen

21 Bilder

0 Literaturstellen

Vervielfältigung und Weitergabe dieser Unterlagen sowie Mitteilung ihres Inhalts an Dritte, auch auszugsweise, nur mit Genehmigung ☒ des DLR ☒ des Auftraggebers.

DLR

**Institut für Aerodynamik und Strömungstechnik
Bunsenstraße 10
37073 Göttingen
Germany
Abteilung AS-EXV**



Anleitungen für das TSP-Messsystem am European Transonic Windtunnel (ETW)

Kurzfassung:

Dieser Bericht fasst die Anleitungen für das Temperature Sensitive Paint Messsystem (TSP-System) am ETW zusammen, die im Rahmen des Projekts „Optimierung des TSP-Systems hinsichtlich Genauigkeit und Effizienz bei industrieller Anwendung im ETW“ (Nr. 3010245) erstellt wurden. Der Bericht enthält Checklisten für TSP-Messungen am ETW, Schritt-für-Schritt-Anleitungen sowie Bedienungsanleitungen für die einzelnen Software-Elemente. Die beschriebenen Programme sind insbesondere ETWin, NI-Pulser, PixelflyCtrl, nt_collect, nToPas und SamTexCtrl.

DEUTSCHES ZENTRUM FÜR LUFT- UND RAUMFAHRT E.V.

Institut für Aerodynamik und Strömungstechnik

Institutsleiter::

(Prof. Dr. Andreas Dillmann)

Abteilungsleiter:

(Dr. Lars Koop)

Verfasser:

(Steffen Risius)

(Dr. Ulrich Henne)

(Dr. Werner Sachs)

Datum:	02.09.2014	Abteilung:	Experimentelle Verfahren	Bericht:	224-2014 A 105
Bearbeitet:	Ilka Micknaus				

Inhalt

Zusammenfassung	2
Abkürzungen	2
1. Einführung	3
2. Inhalt und Aufbau des Berichts	3
3. Kurzanleitung zur 3D-Vorbereitung in nToPas	4
4. Checkliste für TSP-Messungen am ETW	5
4.1. PC5	5
4.2. PC1-4	5
4.3. Linux-PC	6
4.4. Messsystem überprüfen	6
5. Bedienungsanleitungen für Software-Elemente	7
5.1. ETWin: Software als Interface zwischen TSP-Datenerfassung (ToPas) und der RS232-Schnittstelle zum Windkanal (ETW)	7
5.1.1. Einleitung zu ETWin	7
5.1.2. Benutzeroberfläche und Bedienung von ETWin	8
5.1.3. Einstellungen in ETWin	9
5.1.4. Messablauf mit ETWin	9
5.2. Pixelfly ControlSoftware (PF_Ctrl) zur Ansteuerung einer pco.pixelfly-Kamera ..	11
5.2.1. Einleitung zu PF_Ctrl	11
5.2.2. Benutzeroberfläche und Bedienung von PF_Ctrl	11
5.2.3. Tastaturkürzel in PF_Ctrl	12
5.2.4. Einstellungen in PF_Ctrl	12
5.2.5. RemoteControl bei PF_Ctrl	14
5.2.6. Erfassungsdaten von PF_Ctrl	15
5.2.7. Fehlerbehandlung bei PF_Ctrl	15
5.2.8. Systemvoraussetzungen für PF_Ctrl	16
5.2.9. Sonstiges zu PF_Ctrl	16
5.3. Pulserprogramm für die NI PCI-6036E Karte (NI-Pulser)	17
5.3.1. Einleitung zu NI-Pulser	17
5.3.2. Benutzeroberfläche und manuelle Bedienung von NI-Pulser	17
5.3.3. Tastaturkürzel bei NI-Pulser	18
5.3.4. Einstellungen von NI-Pulser	18
5.3.5. RemoteControl Einstellung bei NI-Pulser	19
5.3.6. Hardwarebeschreibung für NI-Pulser	19
Anhang: Englischsprachige Anleitungen	21
Anhang A: Step-by-step manual for TSP measurements at ETW	21
Pictures for step-by-step manual for TSP measurements at ETW	23
Anhang B: Step-by-step manual for nToPas at ETW	25
Pictures for step-by-step manual for nToPas at ETW	29

Zusammenfassung

Dieser Bericht fasst die Anleitungen für das Temperature Sensitive Paint Messsystem (TSP-System) am ETW zusammen, die im Rahmen des Projekts „Optimierung des TSP-Systems hinsichtlich Genauigkeit und Effizienz bei industrieller Anwendung im ETW“ (Nr. 3010245) erstellt wurden. Der Bericht enthält Checklisten für TSP-Messungen am ETW, Schritt-für-Schritt-Anleitungen sowie Bedienungsanleitungen für die einzelnen Software-Elemente. Die beschriebenen Programme sind insbesondere ETWin, NI-Pulser, PixelflyCtrl, nt_collect, nToPas und SamTexCtrl.

Abkürzungen

ETW	European Transonic Windtunnel
ETWin	ETW-Interface steuert die Verbindung zwischen dem Windkanal und dem TSP-System über eine RS232-Schnittstelle und Dateikommunikation
nt_collect	Software zur Datenaufnahme und Speicherung
nToPas	Software zur Auswertung von TSP Daten (new Three dimensional optical Pressure analysis system)
PixelflyCtrl	Software zur Steuerung der pco.pixelfly Kameras
NI-Pulser	Software zur Erzeugung von Pulssequenzen über die National Instruments Karte NI-6036E
SamTex	Kontrolleinheit für ETW Thermoboxen, in der Kameras und LEDs untergebracht sind
SamTexCtrl	Software zur Steuerung der SamTex-Einheiten über die National Instruments Karte PCI-6704 mit Anschlussblock SC-68
TSP	Temperature Sensitive Paint

1. Einführung

Die Temperature Sensitive Paint Messtechnik wurde im Rahmen des Auftrags „Adaption des DLR-cryoTSP-Systems zum Einsatz im ETW“ (Nr. 3005686) in den European Transonic Windtunnel (ETW) integriert und an die ETW GmbH übergeben. Im Rahmen des Folgeauftrags „Optimierung des TSP-Systems hinsichtlich Genauigkeit und Effizienz bei industrieller Anwendung im ETW“ (Nr. 3010245) konnte die Bedienerfreundlichkeit des Messsystems deutlich verbessert werden. Dieser Bericht fasst mehrere Schritt-für-Schritt-Anleitungen zusammen, die im Rahmen des Auftrags (Nr. 3010245) erstellt wurden. Der Wunsch nach dieser Art Kurzanleitungen wurde explizit durch die ETW-Mitarbeiter M. Hauser und M. Czymmek geäußert. Alle Anleitungen wurden nach der Erstellung gemeinsam mit M. Hauser und M. Czymmek besprochen.

Das Ziel der hier vorgelegten Dokumente soll es nicht sein, eine vollständige Bedienungsanleitung für das TSP-System darzustellen. Stattdessen sollen die Dokumente eine Art Gedankenstütze sein, auf die man während der Vorbereitung und Durchführung von TSP-Messungen am ETW zurückgreifen kann. Da sich immer wieder kleine Änderungen im Messablauf am ETW ergeben, kann diese Dokumentation auch nicht als statisch gesehen werden, sondern muss stets den aktuellen Umständen angepasst werden. Da jeder Nutzer andere Elemente als wichtig erachtet wird, könnte auch eine individuelle Ergänzung der Anleitungen durch den jeweiligen Nutzer ggf. hilfreich sein.

2. Inhalt und Aufbau des Berichts

Zunächst wird in Kapitel 3 eine Kurzanleitung für die Vorbereitung eines 3D-Gitters in nToPas gegeben. Im Idealfall wird die Position der Modellmarker im Raum vermessen und dann ihre x,y,z-Koordinaten in nToPas als Markersatz hinterlegt. Sollte dies nicht möglich sein, kann die Vermessung der Modellmarker mit Hilfe von Fotos geschehen, die mit einer Digitalkamera aufgenommen wurden, wie in der Kurzanleitung beschrieben.

Kapitel 4 stellt eine Checkliste dar, die vor und während der TSP-Messung durchgegangen werden sollte.

In Kapitel 5 befinden sich Bedienungsanleitungen für die einzelnen Software-Elemente.

Im Anhang des Dokuments befinden sich zwei ausführliche Schritt-für-Schritt-Anleitungen auf Englisch. Die erste Anleitung erklärt Maßnahmen, die vor einer TSP-Messung durchzuführen sind, während die zweite Anleitung die Auswertung der TSP-Daten mit der Software nToPas beschreibt. Dort ist ebenfalls erläutert, wie eine neue Version von nToPas und des Perl-Moduls installiert werden kann.

Eine ausführliche Dokumentation der einzelnen nToPas-Befehle kann zusätzlich über den Webbrowser aufgerufen werden.

Der Begriff „Fotos“ bezeichnet in diesem Bericht die Fotos, die mit einer Digitalkamera aufgenommen wurden, während der Begriff „Bilder“, die Bilder bezeichnet, die mit den pco.pixelfly-Kameras in der Messstrecke des ETW aufgenommen wurden.

Zur Auswertung der Messung mit nToPas werden vier unterschiedliche Typen von Markern verwendet:

1. Die Modellmarker, die auf dem Windkanalmodell aufgebracht werden.
2. Die 2D-Marker, die im 2D-nToPas auf die Bilder gesetzt werden.
3. Die 3D-Marker, die im 3D-nToPas auf die 3D-Gitterpunkte gesetzt werden.
4. Die Trans-Marker, sind die Verbindung von 3D-Markern und den auf das Bild projizierten 2D-Markern.

3. Kurzanleitung zur Bestimmung von 3D-Markern in nToPas

Die folgende Kurzanleitung beschreibt stichpunktartig, wie die Positionen von Modellmarkern mit Hilfe von Fotos, die mit einer Digitalkamera aufgenommen wurden, vermessen werden können.

1. **Fotos von jpg nach tiff konvertieren:**
convert xxx.jpg -type grayscale -depth 16 xxx.tiff
2. **Fotos von tiff zu i16 konvertieren:**
nt_tiff_2_i16 -src xxx.tiff -dest xxx.i16
3. **Markante Punkte** in den Fotos suchen, um diese als vorläufige 2D-Marker zu verwenden (z.B. Ecken oder Bauteilkanten). Die Positionen dieser Punkte müssen auf dem 3D-Gitter identifizierbar sein:
Im 2D-Topas vier bis fünf vorläufige 2D-Marker setzen und diese abspeichern, z.B. unter xxx_edge_2D.marker
4. **In 3D die gleichen Punkte als 3D-Marker setzen und diese abspeichern, z.B. unter xxx_edge_3D.marker:**
 - a) Zum Finden der Punkte müssen evtl. einige Abstände in der Zeichnung und im 3D-Gitter vermessen oder berechnet werden.
 - b) Die 3D-Marker müssen auf 3D-Gitterpunkten liegen. Daher muss im 3D-nToPas das Gitter ggf. so weit verfeinert werden, bis die 3D-Marker genau genug lokalisiert werden können.
 - c) Die Nummerierung der 2D-Marker und 3D-Marker muss übereinstimmen.
5. **Die Fotos auf das 3D-Gitter projizieren:**
 - a) Folgende Dateien laden: 3D-Gitter (t3d), Fotos (i16), 2D-Marker, 3D-Marker
 - b) Grobes Ausrichten der Trans-Marker und der 2D-Marker
 - c) Durchführen des „Alignments“ mit nToPas: Align, Init, Iterate. Dazu Linsenkorrektur vorläufig abschalten, Fehlerbalken einschalten: „Error Bar“.
 - d) Projizieren der Fotos mit „Get Pixel“ und überprüfen des „Alignments“
 - e) Point of View (POV) speichern.
6. **Überprüfen des „Alignments“:**
 - a) Eine Genauigkeit von 1-2 Pixeln sollte möglich sein, ggf. das 3D-Gitter verfeinern, sollte dieses zu grob sein („Block: Refine x,y“).
 - b) Abstand zum Modell geringfügig ändern (mittlere Maustaste gedrückt halten und die Maus leicht bewegen).
 - c) Bild erneut projizieren („Get Pixel“).
 - d) Schwarze Ränder sollten symmetrisch in x und y um den Flügel herum sichtbar werden.
 - e) Point of View (POV) speichern.
7. Wenn das „Alignment“ stimmt, die **Modellmarker im 3D-nToPas markieren und abspeichern**, z.B. unter xxx_3D.marker.
6. **3D-Blöcke so einteilen**, dass sie den Kameraansichten entsprechen.
 - a) 3D-Marker entsprechend sortieren und zusammenfassen/aufteilen.
9. Sobald das Modell im Windkanal ist, Bilder mit den pco.pixelfly-Kameras machen und die **Zuordnung der Blöcke und 3D-Marker überprüfen**.

4. Checkliste für TSP-Messungen am ETW

Die Punkte 1-3 der folgenden Checkliste sollten vor der TSP-Messung am ETW durchgeführt werden. Danach ist das Messsystem bereit und wartet auf einen Trigger der vom ETW-Messsystem über die RS232-Schnittstelle gesendet wird. Während der Messung muss ständig das Messsystem, wie unter Punkt 4 beschrieben, überprüft werden.

4.1. PC5

- a) "**connect_linux.bat**" ausführen
- b) NI-Pulser starten und Einstellungen überprüfen:
 - Anzahl der Bilder (z.B. "**NoExposures=20** für high speed, **30** für low speed") einstellen
 - Anzahl der Mittellungen ("**Accumulations=1,2 oder 4**") einstellen
 - Zeit zwischen zwei Pulsen einstellen ("**Period=1000ms, 500ms oder 250ms**")
- c) **ETWin starten** und Einstellungen überprüfen
 - ETWin inaktiv lassen (nicht "Activate" klicken)
 - String und Grafik anzeigen lassen
 - Stringdefinition überprüfen und entsprechend den Vorgaben vom ETW einstellen
 - Prüfen ob **RS232-String** kommt (Sonst: Kabelverbindung prüfen, Zuständigen des Bereichs „Test&Data Systems“ informieren)

4.2. PC1 bis PC4

- a) **Stromversorgung** der Kameras anschalten
- b) "**connect_linux.bat**" ausführen
- c) **PFctrl** (Pixelfly Control) starten
- d) **Temperatur** kontrollieren: Temperatur sollte beim Einschalten über 5 Grad Celsius liegen (Sonst: Setpoint der Kameraboxen anpassen, Zuständigen des Bereichs „Test&Data Systems“ informieren)
- e) Kameras auf **LiveView** schalten
- f) Kanalbeleuchtung ausschalten
- g) Kontrollieren, dass der **Kanal dunkel** ist, daher kein Signal auf TSP Kameras (weniger als 30 Counts)
- h) LEDs anschalten, **Schärfe der Aufnahmen** kontrollieren (Sonst: SamTexCtrl starten und "Focus" einstellen)
- i) Bei Kanaltemperatur ungefähr gleich Messtemperatur: Intensität der Kameras prüfen:
 - Step down: maximale Intensität ca. **3200 counts**
 - Step up: maximale Intensität ca. **3600 counts**
 - (Sonst: Belichtungszeit anpassen (**900ms, 400ms oder 150ms**) oder mit SamTexCtrl die "Aperture" einstellen)
- j) LiveView ausschalten
- k) Kamera "Settings" kontrollieren oder einstellen:
 - Anzahl der Bilder (z.B. "**NoExposures = 20** für high speed, **30** für low speed") einstellen, passend zu 1b)
 - Anzahl der Mittellungen ("**Accumulations=1,2 oder 4**") einstellen, passend zu 1b)
 - Interne Bilddivision ausschalten ("**RatioView=None**")
 - Intensität der Kameraansicht anpassen

l) Kameras auf "**Start RC**" (Remote Control) schalten. Danach wird "Stop RC" angezeigt.

4.3. Linux-PC

- a) nt_collect starten: Terminal öffnen "nt_collect" eingeben
- b) Ggf. Einstellungen von nt_collect überprüfen (siehe IB 224-2013A83)
- c) Konfiguration, Serie und Messpunkt (C, S, P) auswählen
- c) Wenn Messbereitschaft erreicht, nt_collect starten: "**Start**" klicken

4.4. Messsystem überprüfen

- a) Messbereitschaft überprüfen:
 - ETWin ist aktiv (**grüne Schaltfläche** mit "Deaktiviere", "Waiting for WT signal" und "Reply R(eady)" erscheint)
 - **RS232-String** wird von ETWin empfangen
 - Kameras sind in **Remote Control** ("Stop RC" wird angezeigt)
 - **NI-Pulser** ist geöffnet
- b) Während der Messung überprüfen:
 - ETWin hat einen **Trigger vom ETW** erhalten (Farbe der Statusanzeige ändert sich und "Reply M" („Measure“) erscheint)
 - **Kameras nehmen Bilder** auf (Statusanzeige: gelb, Bildzähler steigt, Intensität schwankt leicht) (Sonst: Trigger von NI-Pulser nicht erhalten)
- c) Nach der Messung überprüfen:
 - **Kameras speichern Bilder** (Statusanzeige: magenta)
 - Danach gehen Kameras zurück in Bereitschaft (Statusanzeige: grün)
 - nt_collect erhöht den **Messpunktzähler** um eins
 - Terminal in dem nt_collect gestartet wurde gibt **jede Sekunde einen Punkt** aus. Schaltfläche "STOP" wird angezeigt.

5. Bedienungsanleitungen für Software-Elemente

Der folgende Anhang enthält die Bedienungsanleitungen zu den Programmen ETWin, PixelflyControl (PF_Ctrl) und NI-Pulser. Die Programme sind auf den Windows-PCs des TSP-System am ETW installiert und können durch den jeweiligen Icon in der Mitte des Desktops gestartet werden.

5.1. ETWin: Software als Interface zwischen TSP-Datenerfassung (ToPas) und der RS232-Schnittstelle zum Windkanal (ETW)

5.1.1. Einleitung zu ETWin

Das Programm ETWin ist zuständig für die Kommunikation zwischen dem RS232-String, der vom Windkanal gesendet wird, und dem TSP Datenerfassungssystem, welches durch die Software *nt_collect* gesteuert wird. ETWin zeichnet sich insbesondere durch folgende Merkmale aus:

- Einfache und übersichtliche Bedienoberfläche
- Parameter, die während der Messung nicht geändert werden müssen, können in der Initialisierungsdatei mit einem Editor angepasst werden.
- Empfangene Daten können optional in einer Grafik- und Textausgabe in separaten Fenstern dargestellt werden.
- Fensterpositionen und -größen werden bei Programmende gespeichert und beim nächsten Start wieder geladen
- Grafikeinstellungen inklusive der Achseneinteilung werden bei Programmende gespeichert und beim nächsten Start wieder geladen.
- Ausgabe der Daten werden während der Messung in eine ASCII-Datei geschrieben.
- Ständige Erfassung der vom Windkanal gesendeten Daten.
- Wenn *nt_collect* nicht bereit ist oder ein Menü geöffnet ist und somit ein Start der Messung verhindert wird, sendet ETWin „Local“ zurück an den Windkanal über die RS232-Schnittstelle.
- Die Erweiterung um „**command 0**“ beim neuen *nt_collect* wird unterstützt.
- Die einzige Hardwareanforderung an den PC, auf dem die Software läuft, ist das Vorhandensein einer seriellen Schnittstelle (oder USB mit Konverter). Damit kann im Notfall das Programm auch auf einem Notebook laufen.
- Triggerung der Kameras muss extern mit einem extra Programm erfolgen. Damit ist eine größere Hardwareunabhängigkeit gegeben. Die Triggerung kann über die NI-Karte erfolgen oder über den Parallelport (sofern vorhanden)

5.1.2. Benutzeroberfläche und Bedienung von ETWin

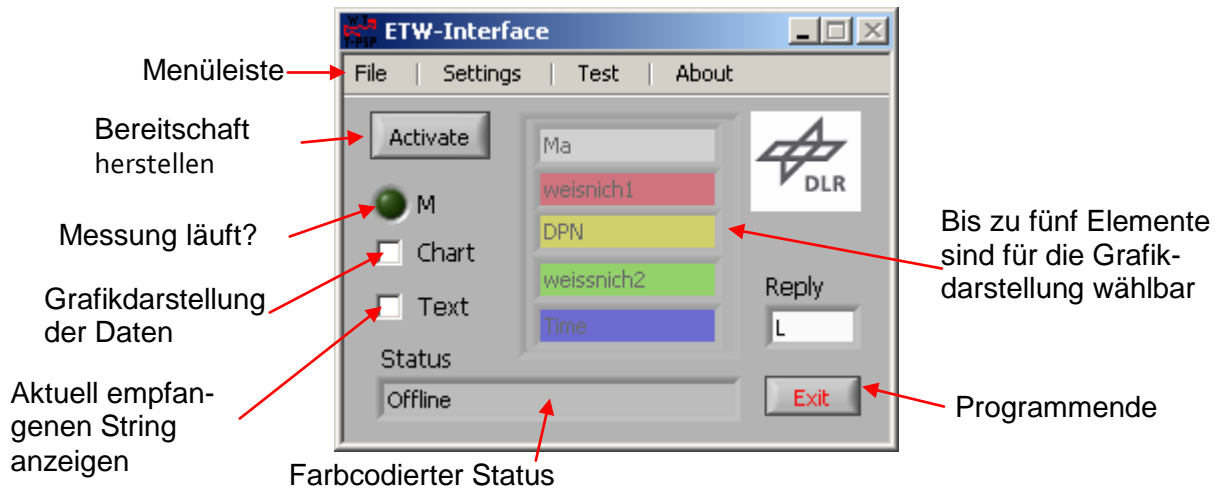


Bild 1: Benutzeroberfläche des Programms *ETWin*

In Bild 1 ist die Benutzeroberfläche des Programms *ETWin* im Status 0 (**Offline**) zu sehen. Der Status wird unten im Hauptfenster farbig codiert angezeigt. Der Button *Activate* dient zum Herstellen der Messbereitschaft (Status 1: **Waiting for WT-Signal**). Dies erfolgt auch bei command 0 von *nt_collect*. Wird ein Trigger vom Windkanal gesendet, so wird die LED *M* aktiviert und Status 2 (**Waiting for cmd 2**) eingenommen. Nachdem Erhalt von **command 2** wird Status 3 (**Waiting for cmd 3**) erreicht. Status 4 (**Waiting for WT data**) wird nur kurz eingenommen, bis nach **command 3** noch ein Datensatz vom Windkanal empfangen wurde. Dann wird Status 5 (**Waiting for wt.rdy**) beibehalten, bis *nt_collect* mit dem Datenpunkt fertig ist. Zum Testen kann über das Menü *Test* → *Simulate WT-Trigger* ein Trigger vom Windkanal simuliert werden. Allerdings ist das Empfangen eines Strings hierfür Grundvoraussetzung. Mit einem Häkchen vor der Box *Chart* wird das Grafikenster mit der Auswahl der Daten vom Hauptfenster dargestellt. Entsprechendes gilt für ein kleineres Textfenster, welches den zuletzt empfangenen String enthält, oder „TIME OUT“, falls innerhalb von 12 Leseversuchen nichts empfangen wurde. Im Grafikenster lassen sich die Skalen automatisch oder manuell einstellen, die Auswahl erhält man mit rechtem Mausklick auf die jeweilige Achse. Zur Anpassung der manuellen Skalierung klickt man auf den oberen oder unteren Wert doppelt und kann dann die gewünschte Zahl eintippen.

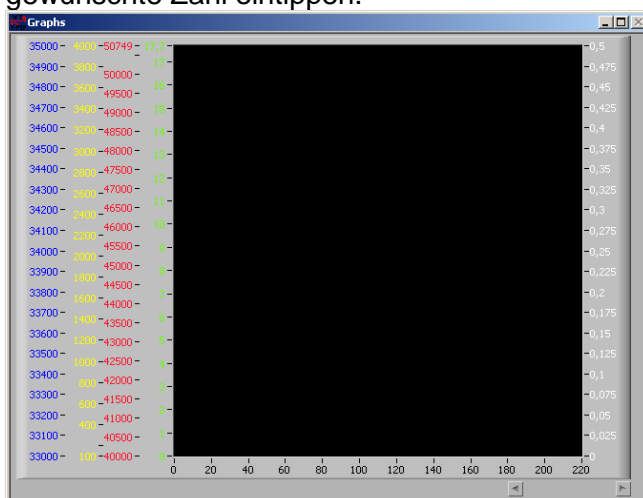


Bild 2: Das Grafikenster

Sämtliche Bedienelemente aller Fenster des Programms zeigen beim Darüberfahren mit der Maus eine kurze Erläuterung ihrer Funktion. Daher wird hier auf eine vollständige Erläuterung der meisten Elemente verzichtet.

5.1.3. Einstellungen in ETWin

Sämtliche Einstellungen werden beim Beenden des Programms in den Standard-Dateien (*General.ini*, *String.ini*, *Graph.ini*, *winpos.ini*), die im gleichen Verzeichnis, wie die Anwendung selbst liegen, gespeichert und beim Starten aus diesen geladen. Gleiches erreicht man zur Laufzeit durch die Menüpunkte *Settings* → *Save* → *All to default* und *Settings* → *Load* → *All from default*

Die Grundeinstellungen werden bei Programmstart aus der Datei *General.ini* gelesen. Über das Menü *Settings* → *Edit* → *General* können die wichtigsten Parameter manuell verändert werden (Bild 2). Diese sichtbaren Einstellungen beziehen sich alle auf die Dateikommunikation mit *nt_collect*. Parameter bezüglich der RS232-Schnittstelle werden nicht angezeigt, denn sie sollten nur von erfahrenen Benutzern direkt in der Datei *General.ini* geändert werden.

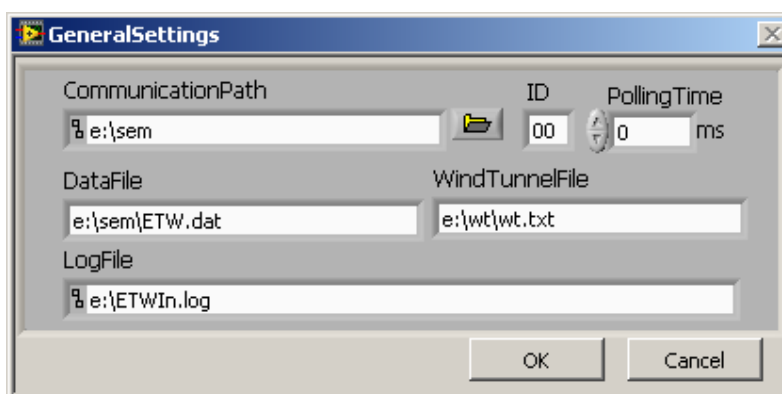


Bild 3: Grundeinstellungen

Die String-Definition wird über das Menü *Settings* → *Edit* → *String* erreicht (Bild 3). Hier können Teilstrings bestimmter Länge definiert werden, die dann zur Auswahl der grafischen Darstellung im Hauptfenster genutzt werden können (sofern sie als Zahlen interpretierbar sind). Weiterhin kann man auswählen, ob die Daten des Triggerstrings in die Windkanaldatei und/oder die Daten während der Bildaufnahme in die Datendatei geschrieben werden sollen. Im unteren Bereich ist die Gesamtlänge aller Teilstrings zu sehen, die nicht länger als der empfangene String sein darf. Ist sie kürzer, werden die überzähligen Zeichen bei der Analyse ignoriert.

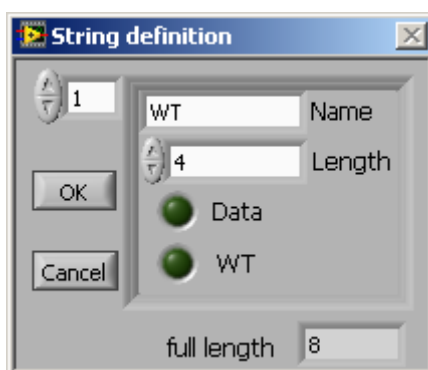


Bild 4: String-Definition

5.1.4. Messablauf mit ETWin

Der zeitliche Ablauf eines Messpunkts (Polare) wird im Folgenden beschrieben:

Falls das Interface nicht aktiviert ist, wird es spätestens beim Start einer Messreihe von *nt_collect* aktiviert. Danach erfolgt die Bestätigung der Bereitschaft mit „R“ als Antwort auf jeden String, der vom Windkanal gesendet wird und mit dem Zeichen 0x0D (ASCII 13=“carriage return“) terminiert sein muss.

Wird im String als erstes Zeichen ein M empfangen (WK-Trigger) ändert sich die Antwort auf „H“ und *ETWIn* startet die Erfassung mit *nt_collect* durch Anlegen der Windkanaldatei (*wt.txt*). Diese enthält diejenigen Daten aus dem gerade empfangenen String, bei denen in den String-Einstellungen der „WT“-Button aktiviert ist. Daraufhin schaltet *nt_collect* mit **command 1** alle Kameras scharf und startet dann mit **command 2** die Triggerung der Kameras und die Speicherung derjenigen Daten in den ab jetzt empfangenen Strings, bei denen in den String-Einstellungen der „Data“-Button aktiviert ist. D.h. in aller Regel, der erste String nach dem Trigger-„M“ wird als erster Datensatz mit Index 000 gespeichert. Bei jedem folgenden String wird der Index um 1 erhöht. Dieser Index wird (3-stellig) direkt an den Bezeichner angehängt, um die einzelnen Datensätze später den Bildern zuordnen zu können. *ETWIn* wartet nun auf **command 3**, welches den Abschluss der Triggersequenz bedeutet. Dann wird der nächste String noch ausgewertet (Antwort ist nun „D“) und als letzter Datensatz angehängt. Weiterhin wird unter dem Namen „DPNS“ noch die Anzahl empfangener Strings gespeichert.

Falls die Triggersequenz schon beendet wird, direkt nachdem der letzte Kamera-Trigger abgeschickt wurde, so ist durch die Übernahme eines weiteren Datensatzes sichergestellt, dass auch für das letzte Bild noch ein Datensatz existiert. Das Kamera-Programm und *ETWIn* geben die Rückmeldung an *nt_collect* erst nach dem Speichern der Bilder bzw. Daten. Die erneute Bereitschaft (Antwort „R“) wird erst wieder signalisiert, wenn *nt_collect* alle Daten erfolgreich verschoben hat, die Windkanaldatei weg ist und eine *wt.rdy*-Datei erzeugt wurde.

5.2. Pixelfly ControlSoftware (PF_Ctrl) zur Ansteuerung einer pco.pixelfly-Kamera

5.2.1. Einleitung zu PF_Ctrl

Für TSP-Messungen wird i.a. eine Reihe von Bildern möglichst schnell hintereinander aufgenommen und abgespeichert. Manuell kann die PCO-Software „CamWare“ für diesen Vorgang verwendet werden. Die Verwendung von PF_Ctrl hat jedoch folgende Vorteile:

1. Die Aufnahme der Bilder und das Speichern erfolgt automatisch und kann über nt_collect gesteuert werden.
2. Um Belichtungszeiten bis ~1 sec zu erreichen, sind die verwendeten Kamerakarten „gepatcht“. Dies führt dazu, dass unter CamWare die Belichtungszeiten nicht korrekt angezeigt werden, sondern um einen Faktor 16 zu klein sind. In der Software PF_ctrl ist dies bereits berücksichtigt, so dass auch bei Einzeltriggerung im „Asynch Shutter“-Modus die Belichtungszeiten richtig angezeigt werden.
3. Durch PF_ctrl wird zu jedem Bild ein Zeitstempel gespeichert, der die zeitliche Zuordnung der Bilder auch ohne externen Trigger zulässt.

5.2.2. Benutzeroberfläche und Bedienung von PF_Ctrl

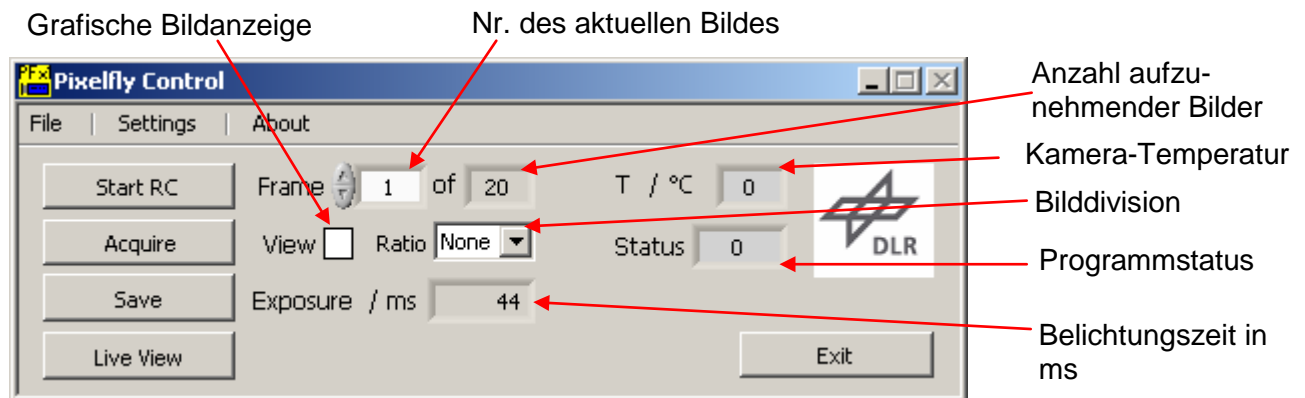


Bild 1: Benutzeroberfläche des Programms **PF_Ctrl**

In Bild 1 ist die Benutzeroberfläche des Programms **PF_Ctrl** zu sehen. Die Grundeinstellungen werden bei Programmstart aus der Datei **Settings.ini** gelesen, die im gleichen Verzeichnis, wie die Anwendung selbst liegt. Über das Menü **Settings**→**Edit** können diese manuell verändert werden.

Die Schaltfläche **Acquire** dient zum Starten/Stoppen der Bilderfassung und die Schaltfläche **Save** zum Speichern oder Abbruch der Speicherung. Gespeichert werden alle Bilder beginnend von 1 bis zum aktuell angezeigten Bild („Frame“).

Mit **Start RC** kann die Schnittstelle zu **nt_collect** gestartet werden (s.u. Remote Control).

Bei aktiviertem **View** wird das zuletzt aufgenommene Bild während der Aufnahme in einem eigenen Fenster angezeigt, wobei die Nummer des angezeigten Bildes unter **Frame** nach erfolgter Erfassung aller Bilder ausgewählt werden kann.

Die Schaltfläche **Live View** erlaubt eine kontinuierliche Aufnahme ohne externe Triggerung, was für Justierungsarbeiten hilfreich ist. Damit man bei häufig anzupassenden Parametern nicht immer über das Menü die Einstellungen ändern muss, sind **Exposure**, **Frame** und **Ratio** im Hauptfenster zusätzlich einstellbar. Die Bedeutung dieser Kontrollfelder wird weiter unten genauer erläutert.

Falls die Verbindung zur Kamera nicht mehr hergestellt werden kann, blinkt das Anzeigefeld für die Kamertemperatur rot.

Der aktuelle Status wird als Nummer angezeigt. Zur schnelleren Wahrnehmung werden die Stati farbig codiert, die im Folgenden erläutert werden.

Grau (0): Idle, warte auf Benutzereingabe (-10): Einstellungen an die Kamera übertragen

Grün (1): RC ist aktiviert, warte auf Dateianforderung zum Start

Orange (2): Warte auf Remote-Anforderung zum Triggern (**command 2**)

Gelb (3): Bildaufnahme läuft, ggf. warte auf externen Trigger

Magenta (5): Speichere Bildsequenz

Blau (6): Warte auf Dateianforderung zum Speichern der Erfassungsdaten.

Weiß (7): Live Modus ist aktiv, es werden kontinuierlich Bilder aufgenommen.

5.2.3. Tastaturkürzel in PF_Ctrl

Folgende Tastenkombinationen sind verfügbar:

Ctrl+E : Editiere Einstellungen

Ctrl+L : Lade Einstellungen

Ctrl+S : Speichere Einstellungen

Ctrl+Q: Programm beenden

Wenn nicht gemessen wird, kann man die Nr. des angezeigten Bildes mit folgenden Tasten ändern:

↑ (Cursor hoch) : +1

↑↑□ (Seite hoch) : +10

↓ (Cursor runter) : -1

↓↓ (Seite runter) : -10

Ctrl+↑□↓ bzw. Ctrl+↑↑□↓↓ : +2/-2 bzw. +20/-20

Die gleiche Änderung erfährt die Belichtungszeit wenn das Programm sich im Live-Modus befindet, wo immer nur das aktuelle Bild (0) angezeigt wird. Wenn allerdings der Fokus auf dem Eingabefeld **Frame** oder **Exposure** liegt, ändert sich mit den Cursortasten der Wert auch ohne Ctrl zu drücken, um den Wert 2.

Achtung: Änderungen der Belichtungszeit im Live Modus werden auch bei nachfolgend gestarteten Messungen beibehalten.

5.2.4. Einstellungen in PF_Ctrl

Die Einstellungen lassen sich über das Menue *Settings->Edit/Load/Save* ändern/laden/speichern und sind in 3 Bereiche unterteilt:

a) Kommunikation mit **nt_collect** (Pfad, „Client“-ID, Dateiname für Erfassungsdaten)

b) Kameraeinstellungen

c) Sonstiges

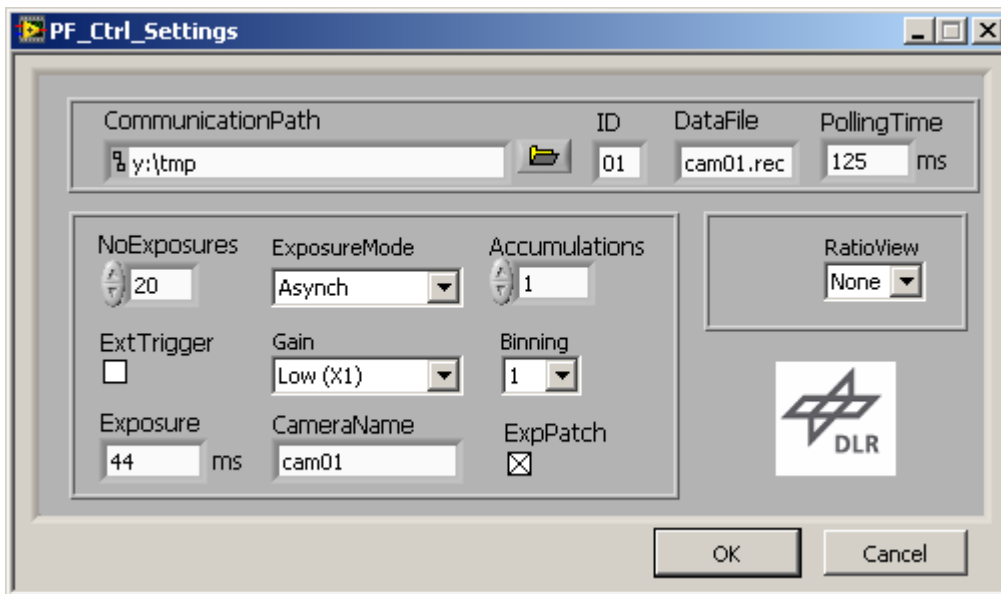


Bild 2: Einstellungen ändern

a) Kommunikation

- **CommunicationPath:** Verzeichnis für **nt_collect**-Kommunikation z.B. "R:\SEM"
- **(Client) ID:** Dateiname ohne Erweiterung für **nt_collect**
 - „01“ bedeutet z.B., dass im Verzeichnis R:\SEM auf eine Datei 01.sync gewartet wird und dann die Datei 01.cmd ausgewertet wird.
- **DataFile:** kompletter Pfad, unter dem die Erfassungsdaten (s.u.) gespeichert werden.
- **PollingTime:** Wartezeit (Intervall), nachdem jeweils auf Existenz der .sync-Datei geprüft wird.

b) Kameraeinstellungen

- **NoExposures:** Anzahl Bilder pro Kamera und Messpunkt (Begrenzt auf 140, wegen Speicherverwaltung)
- **ExposureMode:** Es werden derzeit nur zwei Einstellungen unterstützt:
 - „Asynch Shutter“: Jedes Bild wird einzeln getriggert (*Empfehlung*)
 - „Video Mode“: Es wird eine Sequenz getriggert und die Bilder werden so schnell wie möglich aufgenommen.
- **ExtTrigger:** Wenn ExtTrigger aktiviert ist, wartet die Kamera auf ein TTL Signal, welches z.B. für die Synchronisation mehrerer Kameras erforderlich ist, sonst wird per Software-Trigger die Sequenz oder eine Einzelaufnahme gestartet.
- **Gain:** Verstärkung/Empfindlichkeit des CCD (bei hoher Empfindlichkeit (*High x2*) ist das Signal-Rausch-Verhältnis etwa 1,3-1,4 mal schlechter als bei niedriger Empfindlichkeit (*Low x1*) für ein gleiches Signal.
- **Binning:** Ermöglicht bis zu 2x2 Pixelwerte zu einem einzigen Pixel zusammenzufassen, was die Empfindlichkeit und die Frametransferrate erhöht, jedoch zu einer geringeren Ortsauflösung führt.
- **ExpPatch:** Wenn ExpPatch aktiviert, dann werden die Belichtungszeiten im „Asynch Shutter“-Modus automatisch um einen Faktor 16 korrigiert.
- **Exposure:** Belichtungszeit für die Kamera in Millisekunden.
- **Accumulations:** Anzahl der Einzelaufnahmen, die vom Programm intern addiert als ein einziges Bild dargestellt und abgespeichert werden. Bei bis zu 16 Akkumulationen kann eine Verbesserung des Signal-Rausch-Verhältnisses um fast 4 erreicht werden. Die Darstellung wird immer auf 12bit umgerechnet, aber die gespeicherten Bilder enthalten die aufaddierten Werte (z.B. bei 4 Akkumulationen ergeben sich Werte bis 14bit, d.h. ein Sättigungswert von 16383).

- **CameraName:** CameraName definiert den Dateinamen für die gespeicherten Bilder. Die Bilder werden im Kommunikationsverzeichnis im .b16-Format gespeichert. An den CameraName wird ein „_“ (Unterstrich) und eine 3-stellige laufende Nummer angehängt, beginnend bei 001.

c) Sonstiges

- **RatioView:** hier gibt es 3 verschiedene Einstellungen
 - *none:* Nur Rohbilder zeigen
 - *last:* Dividiere das letzte Bild der Sequenz durch das erste Bild und zeige es an
 - *each:* Dividiere jedes Bild der Sequenz durch das erste Bild und zeige es an
 Hierbei werden die Bilder dividiert und mit 2000 multipliziert, um bei einer 12bit Darstellung (Maximalwert 4095) einen guten Kontrast zu erreichen.

5.2.5. RemoteControl bei PF_Ctrl

Wird die Schaltfläche **RC** aktiviert, so wartet das Programm auf die .sync-Datei und wertet dann die .cmd-Datei aus. Die .cmd-Datei muss dem Format von **nt_collect** genügen, d.h. sie muss in einer Zeile einen Bezeichner gefolgt von einem Wert (durch Leerzeichen getrennt) enthalten. Die Bezeichner sind **phase** und **command**, wobei hier nur **command** ausgewertet wird.

Mit **command 0** kann aus dem Zustand 0 („Idle“) auch in Zustand 1 („RC active“) gewechselt werden. Alle anderen Werte von **command** werden in Zustand 0 mit der Fehlerausgabe „RC inactive“ in die .err-Datei beantwortet.

Im Zustand 1 („RC active“) wird auf 3 mögliche Befehle gewartet:

- command 0:** Anfrage auf Bereitschaft (setzt **RC** aktiv, falls es noch nicht ist) oder Kamera-Einstellungen von der .out-Datei übernehmen, falls sie existiert
- command 1:** Starte Aufnahme, ggf. nach Übernahme der Kamera-Einstellungen, falls die .out-Datei existiert
- command 99:** Gehe in Zustand 7 („Live View“) und speichere das nächste Bild

Auf **command 0**, ohne existierende .out-Datei, wird direkt geantwortet, um zu signalisieren, dass das Programm und die Kamera bereit sind. Mit existierender .out-Datei, werden die Einstellungen in dieser Datei übernommen und PF_Ctrl geht zurück zum Zustand 1. Dies ermöglicht es, Einstellungen auch durch ein externes Programm zu setzen, welches z.B. parallel zu **nt_collect** laufen kann. Außerdem kann ein externes Programm auf die Erfassungsdaten, in der unter Einstellungen definierten Datei (*DataFile*), zugreifen, welche nach **command 0** immer angelegt wird.

Bei **command 1**, mit existierender .out-Datei, werden die Einstellungen in dieser Datei übernommen, dann wird die Kamera gestartet und der Befehl bestätigt (löschen von .sync und erzeugen von .rdy). Ohne existierende .out-Datei, wird direkt die Kamera mit den bestehenden Einstellungen gestartet und der Befehl bestätigt.

Wenn *ExtTrigger* nicht gesetzt ist, dann wird ein Software-Trigger erst nach **command 2** ausgelöst, und im „Asynch Shutter“-Modus auch nachfolgend die Einzelaufnahmen per Software im Programm getriggert, sobald das vorhergehende Bild im Speicher ist. Die Rückgabe an **nt_collect** erfolgt erst, wenn alle Bilder erfasst wurden.

Kommt bei gesetztem *ExtTrigger* ein **command 2**, so gibt es eine Fehlermeldung nach dem Abspeichern der Bilder, da auf TTL-Pulse und nicht auf ein Software-Kommando gewartet wird. Bei **command 3** gibt es dann die Rückmeldung nach erfolgreicher Speicherung der Bilder. Die Bilderfassung endet selbstständig, wenn die Kamera die eingestellte Anzahl von Bildern geliefert hat. Danach beginnt automatisch die Speicherung der Bilder. Die Erfassungsdaten werden erst nach Abspeichern der Bilder geschrieben und danach eine Rückgabe an **nt_collect** geschickt. Während der Erfassung und dem Abspeichern der Bilder ist keine Datei-Kommunikation möglich.

Im Zustand 7 („Live View“) werden (ohne externe Triggerung) dauernd Bilder aufgenommen und angezeigt. Abgespeichert werden die Bilder mit Nummer 000 sowohl nach der Remote-Anforderung, in Zustand 7 zu wechseln, als auch bei jeder weiteren Anforderung von **nt_collect** mit **command 99**. **command 1** mit existierender .out-Datei erlaubt hier nur das Anpassen der Belichtungszeit („Exposure“). **command 0** beendet den „Live“ Modus, das Programm gelangt wieder in den Zustand 1 („RC-Idle“).

Die .out-Datei hat ein ähnliches Format, wie die Datei *DataFile*, die die Erfassungsdaten enthält. Allerdings werden hier die Bezeichner *Accumulations*, *Exposure* und *NoExposures* erwartet, mit denen dann die entsprechenden Werte eingestellt werden können (im Live-Modus nur *Exposure*). Die Bedeutung der Bezeichner stimmen mit denen im Fenster Einstellungen (s.o.) überein. Nachfolgend ist ein typisches Beispiel für eine .out-Datei mit 122 ms Belichtungszeit pro Einzelaufnahme, 30 Bildern und 4 Akkumulationen dargestellt:

```
Exposure      122
NoExposures   30
Accumulations      4
```

Diese Daten werden vom Programm übernommen, wenn eine .cmd-Datei mit **command 0** oder **command 1** gefunden wird.

5.2.6. Erfassungsdaten von PF_Ctrl

Die Datei "*DataFile*" (s.o.) hat folgendes Format (Beispiel):

```
date 07.01.2014
time 15:33:48
camera cam1
res 1024x1384
gain 0
Accumulations 4
NoExposures      30
Exposure 150
Temperature 47
Time1 995
Time2 1800
...
```

Nach Datum und Uhrzeit des Starts folgt der Kameraname, die Pixel-Auflösung (X x Y), die Verstärkung (0=normal, 1=high), die Zahl der für ein erfasstes Bild erforderlichen Einzelbilder (Akkumulationen), die Anzahl der zu erfassenden Bilder, die Belichtungszeit für Einzelbilder (bei 4 Akkumulationen ergeben sich 4 x Belichtungszeit + 4 x Auslesezeit als Minimaler Abstand zwischen zwei akkumulierten Bildern), die Kamertemperatur in °C und die Zeitstempel („Time1“, „Time2“, ...) für alle Bilder in ms gegenüber der Startzeit („time“). Die Zeitstempel geben die Zeit zwischen dem Scharfschalten der Kamera und der Übertragung des jeweiligen Bildes ins RAM an.

5.2.7. Fehlerbehandlung bei PF_Ctrl

Sollte die Kamera nicht angeschlossen sein, hat man die Möglichkeit, die Einstellungen zu ändern, bevor sich das Programm wieder beendet.

Geht während der Messung die Verbindung zur Kamera verloren, so blinkt die Temperaturanzeige rot. Bei allen Remote-Anfragen wird dann immer der Fehler „Camera connection lost“ in die .err-Datei ausgegeben.

Wird die Schaltfläche **RC** gedrückt und es existiert schon eine .sync-Datei, dann erscheint ein Hinweisenfenster, nach der die .sync-Datei ohne weitere Analyse gelöscht wird (Auswahl „Cleanup“) oder der Befehl in der .cmd-Datei gelesen und auch ausgeführt wird (Auswahl „Accept“).

Sollte ein unpassender Befehl empfangen werden, so erscheint ein Hinweisenfenster und ermöglicht ggf. noch den Fehler anderweitig zu beheben (Auswahl „Continue after fixing“) oder auch RC zu beenden (Auswahl „Abort RC“).

Beim Setzen der Parameter durch RC-Befehl 0 oder 1 mit existierender .out-Datei wird bei falschen Werten (1>exp>1020, 1>Accumulations>8 oder 1>NoExposures>140) der Fehler „Parameter out of range“ in die .err-Datei ausgegeben, aber der betroffene Wert trotzdem auf den entsprechenden Grenzwert gesetzt.

Wird im RC-Betrieb während der Bilderfassung diese manuell durch Drücken von **Stop Acquire** gestoppt, so werden alle bis dahin aufgenommenen Bilder gespeichert. Wird beim Speichern im RC-Betrieb **Stop saving** gedrückt, so wird das Speichern abgebrochen. In beiden Fällen wird kein Fehler ausgegeben. Dies ermöglicht es, einen Messpunkt noch sinnvoll zu beenden, falls eine Kamera eine nicht die ausreichende Zahl an Triggerpulsen erhalten hat.

Bricht **nt_collect** während der Erfassung oder des Speicherns den aktuellen Messpunkt ab, so muss ggf. noch einmal **command 0** gesendet werden, um wieder in Zustand 1 („RC Idle“) zu gelangen und Messbereitschaft herzustellen.

Ein typischer Fehler bei der Konfiguration von **nt_collect** besteht darin, den Befehl 3 nicht zu senden, auf den das Kameraprogramm nach dem Speichern der Bilder wartet (Status 6 (blau)). Wird also Status 6 länger angezeigt, so wartet das Programm auf den Befehl 3, der offensichtlich in **nt_collect** für diesen Client nicht konfiguriert ist, oder die Erfassung wurde in **nt_collect** abgebrochen.

5.2.8. Systemvoraussetzungen für PF_Ctrl

4GB Ram, Pixelfly Kamera und Framegrabber-Karte
Labview Runtime Engine 9.0, Windows XP oder Windows 7

5.2.9. Sonstiges zu PF_Ctrl

Es wurde bewusst für die Bildanzeige auf eine komfortablere Bilddarstellung mit dem Vision-Paket verzichtet, da hierfür auch extra Runtime-Lizenzen für jedes System, auf dem es läuft, erworben werden müssten.

Bei externer Triggerung können die Belichtungszeiten frei gewählt werden. Die Periode für einen externen Pulsgenerator muss immer größer als 139 ms sein.

5.3. Pulserprogramm für die NI PCI-6036E Karte (NI-Pulser)

5.3.1. Einleitung zu NI-Pulser

Bei Messungen mit mehreren CCD-Kameras wird i.a. eine Reihe von Bildern möglichst schnell hintereinander aufgenommen, wobei die Bilder der verschiedenen Kameras möglichst gleichzeitig erfasst werden sollen. Dies lässt sich durch externe Triggerung erreichen. Im Folgenden wird ein Programm beschrieben, welches sich zum Einsatz mit der Datenerfassungssoftware **nt_collect** eignet. Es startet bei Bedarf eine Sequenz, bestehend aus einem Spannungspegel (5 V, z.B. um eine LED zu schalten), sowie aus Pulsen an zwei Ausgängen, einen für Kamerabelichtung und einen für die Ansteuerung von Blitzlampen.

5.3.2. Benutzeroberfläche und manuelle Bedienung von NI-Pulser

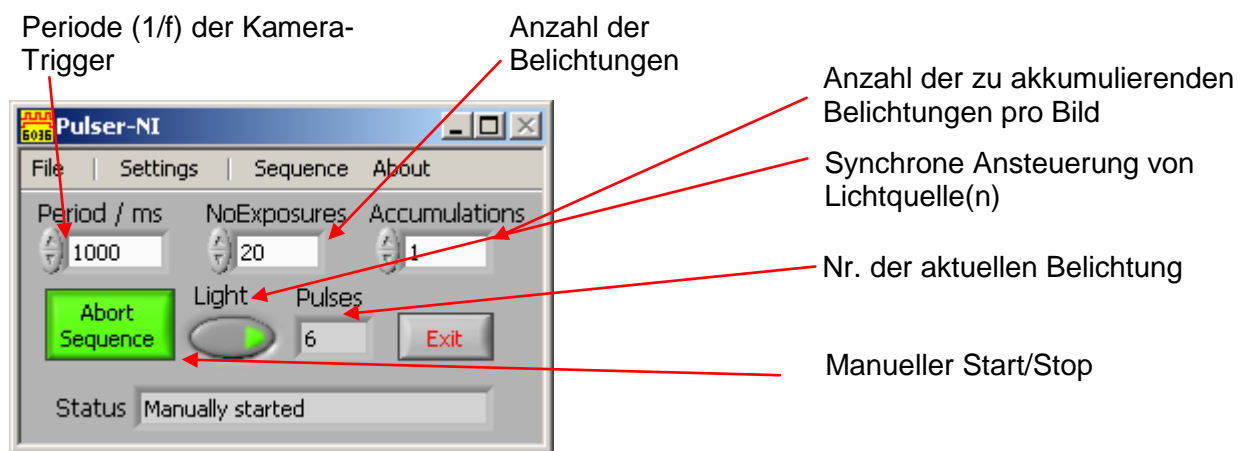


Bild 1: Benutzeroberfläche des Programms NI-Pulser

In Bild 1 ist die Benutzeroberfläche des Programms NI-Pulser zu sehen. Die Grundeinstellungen werden bei Programmstart aus der Datei *Settings.ini* gelesen, die im gleichen Verzeichnis, wie die Anwendung selbst liegt. Über das Menü **Settings**→**Edit** können diese manuell verändert werden.

Die Schaltfläche *Sequence* dient zum Starten/Stoppen der Sequenz. Sie ist (wie in Bild 1 zu sehen) hellgrün unterlegt, wenn die Sequenz läuft. Ist die Schaltfläche *Light* deaktiviert, werden nur Kamera-Trigger ausgegeben. Beim Programmstart ist sie immer inaktiv. Bei manuellem Start wird die Schaltfläche *Light* nicht geändert, weder beim Sequenzstart, noch beim Ende. Dies muss manuell vor einem Sequenzstart geschehen. Bei aktiviertem *Light* ist der Ausgang **AO0** auf 5 V (high) gesetzt.

Damit man bei häufig anzupassenden Parametern nicht immer über das Menü die Einstellungen ändern muss, sind *NoExposure*, *Period* und *Accumulations* im Hauptfenster zusätzlich einstellbar. Ihre Funktion wird in Abschnitt 3 (Einstellungen) genauer erklärt. Die Anzeige *Pulses* ist nur sichtbar, wenn eine Sequenz läuft und zeigt die Zahl der bisher erzeugten Kamera-Trigger an. Da pro Bild durch Verwendung von *Accumulations* (>1) auch mehrere Kamera-Trigger erfolgen können, kann die Anzeige auch Werte bis zu Vielfachen von „NoExposures“ annehmen.

Die Anzeige *Status* kann folgende Werte annehmen, die den Status des Programms definieren: *Idle*, *Manually started*, *Remotely started*.

5.3.3. Tastaturkürzel bei NI-Pulser

Folgende Tastenkombinationen sind verfügbar:

Ctrl+E: Editiere Einstellungen

Ctrl+L: Lade Einstellungen

Ctrl+S: Speichere Einstellungen

Ctrl+A: Start/Stopp der Sequenz

Ctrl+Q: Programm beenden

5.3.4. Einstellungen von NI-Pulser

Die Einstellungen lassen sich über das Menu *Settings->Edit/Load/Save* ändern/laden/speichern und sind in 2 Bereiche unterteilt:

a) Kommunikation

b) Sequenz

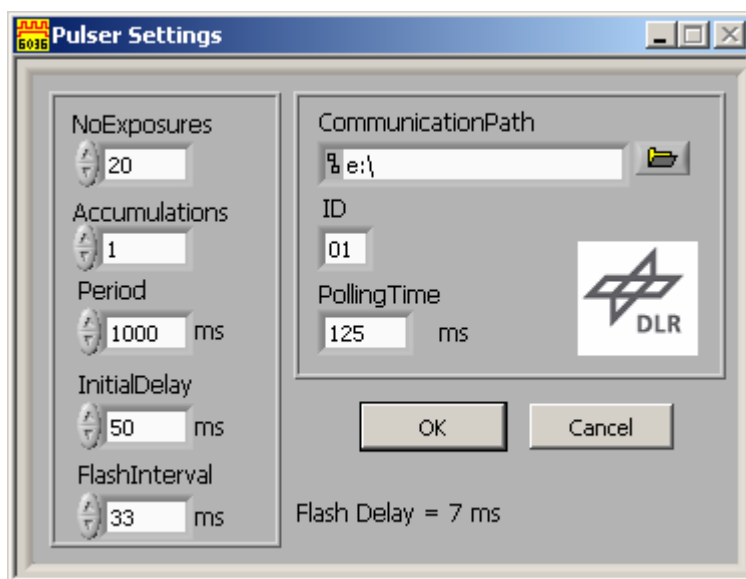


Bild 2: Einstellungen ändern

a) Kommunikation

- **CommunicationPath:** Verzeichnis für Datei-Kommunikation mit **nt_collect** z.B. "R:\sem"
- **(Client) ID:** Dateiname ohne Erweiterung für **nt_collect**
 - „01“ bedeutet z.B., dass im Verzeichnis R:\sem auf eine Datei 01.sys gewartet wird und dann die Datei 01.cmd ausgewertet wird.
- **PollingTime:** Wartezeit (Intervall), nachdem jeweils auf die Existenz der .syc-Datei geprüft wird.

b) Sequenz

- **NoExposures:** Anzahl zu speichernder Bilder pro Kamera und Messpunkt
- **Accumulations:** Anzahl der Belichtungen, die vom Kameraprogramm intern addiert und als ein Bild dargestellt und abgespeichert werden. Die Zahl N der tatsächlichen Kamera-Trigger ist:
$$N = \text{NoExposures} \times \text{Accumulations}$$
- **Period:** Zeit zwischen zwei Kamera-Trigger
- **InitialDelay:** Wartezeit zwischen Setzen des Ausgangs **A00** auf 5 V (high) und dem ersten Kamera-Trigger im RC-Betrieb mit **command 2**
- **FlashInterval:** Zeit zwischen zwei Pulsen der Blitzlampe innerhalb einer Belichtung
- **FlashDelay:** Die Zeit zwischen Kamera-Trigger und erstem Blitz ist fest auf 7 ms eingestellt.

5.3.5. RemoteControl Einstellung bei NI-Pulser

Das Programm wartet im Zustand *Idle* auf eine *.syc*-Datei und wertet dann die *.cmd*-Datei aus. Die *.cmd* Datei muss dem Format von **nt_collect** genügen, d.h. sie muss in einer Zeile einen Bezeichner gefolgt von einem Wert (durch Leerzeichen getrennt) enthalten. Die Bezeichner sind „**phase**“ und „**command**“.

Es werden 4 mögliche Befehle akzeptiert:

command 0: Anfrage auf Bereitschaft

- falls keine *.out*-Datei existiert: Rückmeldung ohne Aktion
- Falls eine *.out*-Datei existiert, werden die darin gefundenen Einstellungen übernommen

command 1:

- falls keine *.out*-Datei existiert: Rückmeldung ohne Aktion
- Falls eine *.out*-Datei existiert, werden die darin gefundenen Einstellungen übernommen
 - o Bei **phase**=0 wird der Ausgang **AO0** auf 0 V gesetzt, bei **phase**=1-3 auf 5 V.

command 2: Start der Sequenz.

- o Bei **phase**=0 wird der Ausgang **AO0** auf 0 V gesetzt, sonst auf 5 V. Nach einer Wartezeit von *InitialDelay* ms wird die Sequenz gestartet.
- o Bei **phase**=0 wird ohne Pulse auf **Ctr1** (Blitzlampen-Trigger) gemessen, sonst wird mit **Ctr1** gemessen. Am Ende der Sequenz wird der Ausgang **AO0** wieder auf 0 V gesetzt.

command 3,4: Der Ausgang **AO0** wird unabhängig vom Wert von **phase** auf 0 V gesetzt.

In jedem Fall erfolgt die Rückmeldung durch Erzeugen einer *.rdy*-Datei nach Abschluss der Aktion.

Die *.out*-Datei hat ein ähnliches Format, wie die *.cmd*-Datei. Allerdings werden hier nur die Bezeichner *Accumulations*, *Period* und *NoExposures* unterstützt, mit denen dann die entsprechenden Werte eingestellt werden können. Das ist von Vorteil für eine vollständige Einstellung von Zeiten über ein zentrales Programm, welches die entsprechenden 3 Parameter auch für die Kameras einstellen könnte. Bei Verwendung von pco.pixelfly-Kameras ist darauf zu achten, dass der Wert von *Period* mindestens der maximalen Kamera-„*Exposure*“ plus 90 ms (Auslesezeit) entspricht.

5.3.6. Hardwarebeschreibung für NI-Pulser

Es werden 3 Ausgänge einer NI PCI-6036E Karte angesteuert, welche TTL-Pulse bzw. Pegel von 5V ausgeben. Dabei sind die Ausgänge wie folgt belegt, die auch im „About“-Fenster nachlesbar sind.

<u>Ausgang</u>	<u>Funktion</u>
Ctr0	Blitzlampe
Ctr1	Kamera
AO0	Ausgang für LEDs/Licht

Wird eine Anschlussbox vom Typ BNC2110 von NI verwendet, kann man in Bild 3 die Belegung der Ausgänge ablesen.

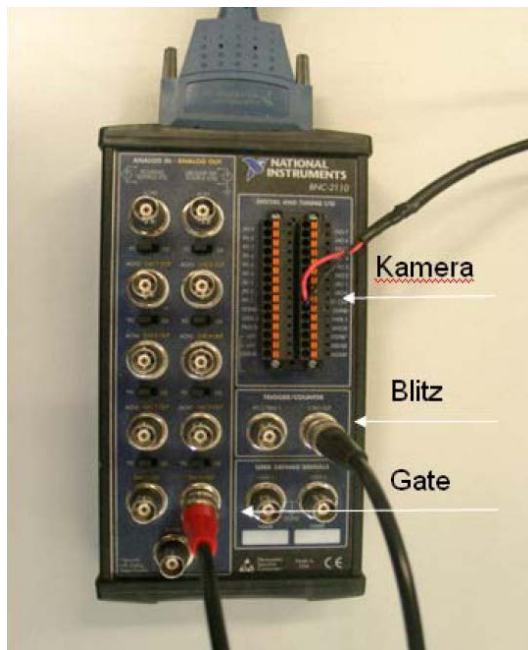


Bild 3: Belegung der Ausgänge an der Anschlussbox BNC 2110

Anhang: Englischsprachige Anleitungen

Die folgenden englischsprachigen Anleitungen erklären das Vorgehen bei TSP Messungen am ETW. Sie sind noch ausführlicher als die Kurzanleitung von Kapitel 3 und die Checkliste von Kapitel 4. Da die offizielle Betriebssprache am ETW Englisch ist, wurden diese Anleitungen auch auf Englisch verfasst, so dass ggf. jeder Mitarbeiter sich an Hand dieser Anleitungen einen Überblick über das Vorgehen bei TSP Messungen verschaffen kann.

Anhang A: Step-by-step manual for TSP measurements at ETW

1. Connections

1.1. Hardware connections:

- 1.1.1. Connect the 4 Cameras to the Windows PCs 1-4
- 1.1.2. Connect the 4 SamTex units to the first NI-Card of PC5 (PCI-6704 with block SC-68)
- 1.1.3. Connect the trigger cables from the second NI-Card (NI-6036E) to the cameras

1.2. Software connections:

- 1.2.1. Connect all PCs to the Linux PC with "**connect_linux.bat**"

2. SamTexCtrl

2.1. Start SamTexCtrl on PC5

2.2. Check correct connection of the cameras with the SamTex units

- 2.2.1. Switch the according camera to "Live View" and select "View" to display the image preview.
- 2.2.2. Select a voltage between 3 and 6 V for the SamTex units.
- 2.2.3. Try all SamTex units and adjust their parameters (Aperture, Focus, Roll, Pitch). Check whether you can observe the expected change at the cameras.

2.3. Adjust the view of the cameras:

- 2.3.1. Place the model in the test section
- 2.3.2. Position the field of view of the cameras on the TSP areas of interest
- 2.3.3. Keep in mind that the angle of attack may change and the TSP areas of interest may leave the field of view

2.4. Adjust the focus of the cameras:

- 2.4.1. Open the aperture as far as possible in order to reduce the focal depth. (Maybe it is necessary to reduce the exposure time.)
- 2.4.2. Adjust the focus of the cameras

2.5. Adjust the aperture of the cameras:

- 2.5.1. Switch the LED light sources to "ON"
- 2.5.2. Select a representable exposure time (maximum 150ms)
- 2.5.3. Close the aperture until the maximum intensity in the image is below **2000** counts for cryogenic tests or below **3300** counts for ambient tests, in order to avoid oversaturation of the cameras. (The TSP fluorescence intensity will increase with decreasing temperature.)

2.6. Optionally close SamTexCtrl in order to avoid unwanted parameter changes.

2.7. Monitor the camera temperature all the time.

3. PFCtrl (PixelFlyControl)

3.1. Start PFCtrl on all Windows PCs 1-4

3.2. Check the connections with the cameras

- 3.2.1. If PFCtrl gives an error message and stops working a connection to the camera could not be established
- 3.2.2. If the temperature field is blinking red, the connection to the camera was lost

3.3. Check the settings of PFctrl (picture 1)

3.4. "ID", "DataFile" and "CameraName" should be set at the same number as the camera PC.

- 3.5. Check the intensity of the image:
 - 3.5.1. Select the yellow cursor and move it around the window. At the bottom right field below the preview image you can read off the intensity.
 - 3.6. Adjust the exposure time:
 - 3.6.1. When the measurement temperature range is reached adjust the exposure time of the cameras such that the maximum intensity is below 3300 counts, in order to avoid oversaturation of the cameras. (Exposure time should be about 20 to 40 ms at 130K but varies strongly between tests.)
 - 3.7. Start remote control by pressing "Start RC" for all cameras.
 - 3.8. When the measurement temperature is reached, adjust the exposure time again, so that the maximum intensity is below 3300 counts.
 - 3.9. Examples for exposure times: Ratio=Each -> 2500-3500 counts; Ratio=Last -> 2500-3500 counts; Ratio=None -> #Averages*3500=Intensity (12000-15000).
 - 3.10. In case a camera fails, restart the program.
 - 3.11. In case of communication errors: Shut down all programs, stop nt_collect, delete all communication files, restart all programs.
4. **ETWin**
 - 4.1. Start ETWin on PC5 and check the settings (picture 2)
 - 4.2. Display the chart and select the parameters you want to monitor. (For example: TMrear=130-140K, Alpha=-2-5, TMnose=30-40, Tflow=130-140, Mach=0-0.8.)
 - 4.3. Display the string by selecting "Text" in order to check whether a string is send from ETW.
 - 4.4. Select in the settings menu "Save all to default"
5. **Pulser**
 - 5.1. Start Pulser on PC 5 and check its settings (picture 3)
6. **nToPas**
 - 6.1. Please check the "**Step by step manual for nToPas at ETW**" for detailed descriptions.
 - 6.2. Create a project directory, using nt_create_project
 - 6.3. Start nToPas on the LINUX PC by opening a terminal and typing: "**nt_com**".
 - 6.4. Put all necessary data into the project directory (marker files, RC-file, bib-file, batch-files)
 - 6.5. Load a picture and set the markers as start markers for each view
 - 6.6. Check the location of the intensity correction for different images (TMx, TMy)
 - 6.7. Conduct a test measurement in Configuration 99 and check whether all files are saved at the correct location
7. **nt_collect**
 - 7.1. Start nt_collect on the LINUX PC by opening a terminal and typing: "**nt_collect**"
 - 7.2. Check the settings of nt_collect (picture 4 and 5)
 - 7.3. Select the correct Configuration, Series and Point (C,S,P)
 - 7.4. Start nt_collect by pressing "Start"
8. **Conduct a test measurement**
 - 8.1. Select Configuration 99 in nt_collect as a test configuration.
 - 8.2. Make all devices ready for measurement (nt_collect:"Start", ETWin: "Activate", PFCtrl: "Start RC", Start NI-Pulser)
 - 8.3.
 - 8.4. Start the measurement
 - 8.4.1. Option a: Go to ETWin and click on "Test"+ "Simulate WT-Trigger". The next string will start a measurement.
 - 8.4.2. Option b: Trigger the measurement by sending "M" via the RS232 string from ETW.

Pictures for step-by-step manual for TSP measurements at ETW

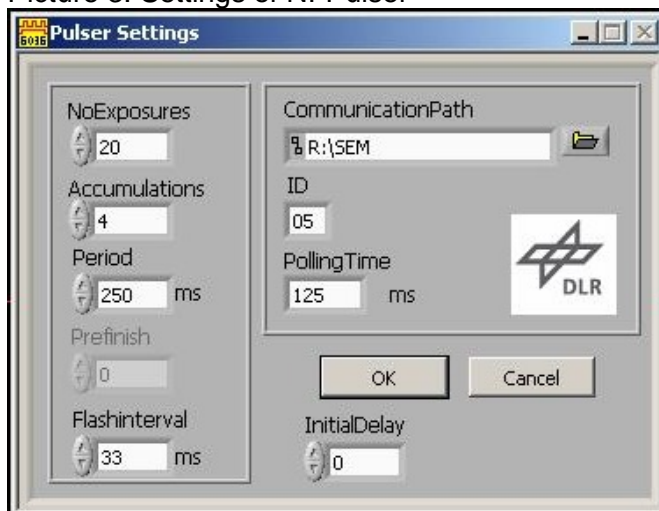
Picture 1: Settings of PFCtrl



Picture 2: Settings of ETWin



Picture 3: Settings of NI-Pulser



Picture 4: Configuration of nt_collect

Configuration

Moving Rules

No	Dir-Name	File-Expression	use
0	TSPdat	tsp.dat	<input checked="" type="checkbox"/>
1	cam1	cam01*.b16	<input checked="" type="checkbox"/>
2	cam2	cam02*.b16	<input checked="" type="checkbox"/>
3	cam3	cam03*.b16	<input checked="" type="checkbox"/>
4	cam4	cam04*.b16	<input checked="" type="checkbox"/>
5	CamInfo	*.rec	<input checked="" type="checkbox"/>
6	0	0	<input type="checkbox"/>
7	0	0	<input type="checkbox"/>
8	0	0	<input type="checkbox"/>
9	0	0	<input type="checkbox"/>
10	0	0	<input type="checkbox"/>
11	0	0	<input type="checkbox"/>
12	0	0	<input type="checkbox"/>
13	0	0	<input type="checkbox"/>
14	0	0	<input type="checkbox"/>
15	0	0	<input type="checkbox"/>

Directories

Project Dir:

Com. Dir:

WT-File:

Other Definitions

Sleep Time: Averages:

#PC's:

Picture 5: PC Commands of nt_collect

PC Commands

Functionality

PC	0	1	2	3	4
0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
1	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
2	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
3	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
4	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
5	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
6	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
7	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
8	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
9	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Anhang B: Step-by-step manual for nToPas at ETW

1. Copy a prepared project directory from USB to the directory “/data/Projects”:

- 1.1. Open a terminal and type: **“rsync -av /media/.../month.ETW.project_name /data/Projects/year/month.ETW.project_name”**.

2. Start and check the settings of nToPas

- 2.1. Open a terminal on the Linux PC and type **“nt_com”**. nToPas will start.
- 2.2. Check the correct window positions:
 - 2.2.1. Start **“2D”** and **“2D-GUI”** and check, whether the windows are displayed on the correct monitor (Picture 1).
 - 2.2.2. If the windows are not displayed on the correct monitor you can change their location with the **“Pref”**-button (Picture 2).
 - 2.2.3. Select the menu for which you want to change the position in the nt-commander and then select the monitor with **“Shell-Displays”**. Recommended is the setting of **“0”**
 - 2.2.4. The position of shells can be defined with **“-geometry wxh+x+y”** with w,h in units of characters and x,y in pixels from upper left corner
- 2.3. Set the data root directory:
 - 2.3.1. Open the file selector in the 2D-GUI by selecting **“File”**.
 - 2.3.2. Select **“Set Data Root”** and select the project directory, as for example: **“/data/Projects/year/01.ETW.project_name/ToPas”**.
- 2.4. Define the project directory as a bookmark in the file selector:
 - 2.4.1. Open the file selector in the 2D-GUI by selecting **“File”**.
 - 2.4.2. To load an i16 file, select **“Load”** (Picture 3).
 - 2.4.3. In the file selector select **“Pref”** (Picture 4).
 - 2.4.4. Enter a new bookmark by typing **“dir /Project_name/ToPas Project_name”** in the text window and save it (Picture 5).
 - 2.4.5. A new bookmark will appear in the file selector.
- 2.5. Set the 2D window size:
 - 2.5.1. In the 2D-GUI adjust the window size. Recommended is a size of 100%. Because with a window size of 100% misalignment of selections should be avoided.
 - 2.5.2. In order to store the current window position and size, hover the mouse over the image window and press **“s”**.
- 2.6. Check the color map settings:
 - 2.6.1. Select **“CMap”** in the 2D-GUI. (Picture 6)
 - 2.6.2. Check the upper and lower bounds of the color map (**“ColorMap Bands”**) and adjust them to your desired settings.
 - 2.6.3. For an export of pictures a lower band with a threshold may be useful.
 - 2.6.4. Check whether certain band widths are marked.
 - 2.6.5. You can adjust the color map automatically with the following steps:
 - 2.6.5.1. Move your mouse to the 2D window (where the image is displayed).
 - 2.6.5.2. Press the letter **“u”** on the keyboard.
 - 2.6.5.3. The color map will be adjusted automatically.
- 2.7. Check dynamic functions:
 - 2.7.1. Select **“Dyn”** from the 2D-GUI and make sure that the following functions are loaded: **adaptSRCmark2.so, checkmarkers.so**
- 2.8. Check the batch and script settings:
 - 2.8.1. Select **“Batch”** from the 2D-GUI.
 - 2.8.2. In the Batch window select the correct data root directory: **“Data Root”**. The data root directory is inside the project directory, for example: **“/data/Projects/2014/01.ETW.HINVA/ToPas”**.
 - 2.8.3. Select the correct batch files:
 - 2.8.3.1. Select **“Set 2D”**.

- 2.8.3.2. Use the file selector to go to the project directory and select "project_name/ToPas/Batch/V_2D.b0" for the 2D batch files.
- 2.8.3.3. Hint: if the file selector does not display all files correctly, check the **filter settings** at the bottom of the file selector.
- 2.8.3.4. The correct batch scripts should now be loaded. You can open individual scripts by selecting them with the right mouse button. Or you can open all scripts at once by using the "E"-button on the right.
- 2.8.3.5. The library which contains most of the sub-functions is called "ETW_TSP_lib.pl" and is located at "project_name/ToPas/Batch/ETW_TSP_lib.pl".
- 2.8.4. Check the configuration, series and point (C,S,P) settings in the batch window. At the beginning of the test the C,S,P should be set to 1,1,1.
- 2.9. Save the **configuration** file of the project:
 - 2.9.1. Select "Files" in the 2D-GUI.
 - 2.9.2. Select "Save" at the bottom of the window (located above "Config+DataRoot").
 - 2.9.3. Select the path and name under which you want to save the **configuration file**, for example in the home folder: "/home/HINVA_2D.conf"

3. Load an image (i16) file to nToPas

- 3.1. Convert b16 camera images to i16 images:
 - 3.1.1. Open a terminal and go to the directory where the image is located. For example: "cd /data/pccom/sem" (this is the directory where the cameras normally save the images.)
 - 3.1.2. Convert a single image with the terminal command: "nt_b16_2_i16 -src CAM1_01.b16 -dest CAM1_01.i16".
- 3.2. Start nToPas
 - 3.2.1. Open a terminal and type: "nt_com".
 - 3.2.2. Select "Files".
 - 3.2.3. Select "Load".
 - 3.2.4. Select the image you want to open: "/data/pccom/sem/CAM1_01.i16".
 - 3.2.5. If necessary adjust the color map:
 - 3.2.5.1. A) **Automatically**: Move the mouse over the 2D image and press "u" for automatic adjustment.
 - 3.2.5.2. B) **Manually**: Open the color map dialog by selecting "CMap" and perform individual adjustments of the color map.

4. Set or load the 3D markers and save transformed markers

- 4.1. Start nToPas and select "3D" and "3D-GUI".
- 4.2. Load the necessary files in the "Files" menu from top to bottom: Select "Files" in the 3D-GUI.
 - 4.2.1. Load the grid file (t3d) in the project directory, for example: "/data/Projects/year/project_name/C01/3D/CAM1.t3d".
 - 4.2.2. Load the image file (i16), for example from: "/data/pccom/sem/CAM1.i16". (This is the directory where the cameras save the images on default. The b16 images from the cameras have to be converted to i16 images. See step 5.1: Conversion of b16 to i16 images.)
 - 4.2.3. A) If 3D markers have **not** been defined: **set 3D markers** at the location, where you assume they should be located:
 - 4.2.3.1. Select "Marker" in the 3D-GUI and select "Add".
 - 4.2.3.2. Select the grid point where you assume the marker should be located.
 - 4.2.4. B) If 3D markers have been defined before: **Load the 3D markers** with "Files" menu from: "/data/Projects/year/project_name/C01/3D/CAM1.marker"
 - 4.2.5. Save the **transformed 3D markers** (this is a 2D marker set) by opening the "Files" menu and selecting "Save T.M." and save markers for example as: "/data/Projects/year/project_name/C01/Etc/CAM1.marker". (This will generate a set

of 2D markers which have the same numeration as the 3D markers. The set of 2D markers can be used as the 2D start markers.)

5. Load and adjust 2D markers

- 5.1. Start nToPas with the terminal command "nt_com".
- 5.2. Select "**2D**" and "**2D-GUI**".
- 5.3. Select "**Marker**" in the 2D-GUI.
- 5.4. Load a camera image by selecting "**Load Image**", for example:
"/data/pccom/sem/CAM1_01.i16".
- 5.5. Load 2D markers, by selecting "**Load SRC**", for example:
"/data/Projects/year/project_name/C01/Etc/CAM1.marker".
- 5.6. Copy source to destination markers with "**src->dest**".
- 5.7. If necessary insert new markers with "**Insert**" and define marker number on the right and select the position in the image window.
- 5.8. If necessary delete markers with "**Delete**" and select the marker in the image window.
- 5.9. Use "**Move Align**" to move the markers on top of the black dots in the image.
 - 5.9.1. Select "Move Align".
 - 5.9.2. Select a marker in the image.
 - 5.9.3. Select the new position of the marker in the image.
 - 5.9.4. Terminate the command by clicking with the right mouse button on the image.
- 5.10. Save destination markers by selecting "**Save DEST**".
- 5.11. Copy the destination markers to the source markers by selecting "**dest->src**".
- 5.12. Improve the marker position by selecting "**Improve**" and define the rectangle size in both text fields on the right of the "Improve" command to be about 11 (amount of pixels, depending on the marker sizes).
- 5.13. Copy the new source markers to destination markers "**src->dest**".
- 5.14. Save the dest markers with "**Save DEST**", as
"/data/Projects/year/project_name/C01/Etc/CAM1_start.marker".

6. Check and adjust 3D markers

- 6.1. Start nToPas and select "**3D**" and "**3D-GUI**".
- 6.2. Load the necessary files in the "Files" menu from top to bottom: Select "**Files**" in the 3D-GUI.
 - 6.2.1. Load the grid file (**t3d**) in the project directory, for example:
"/data/Projects/year/project_name/C01/3D/CAM1.t3d".
 - 6.2.2. Load the image file (**i16**), for example from: "/data/pccom/sem/CAM1.i16". (This is the directory where the cameras save the images on default. The b16 images from the cameras have to be converted to i16 images. See step 5.1: Conversion of b16 to i16 images: "**nt_b16_2_i16 -src src_file_name -dest dest_file_name**".)
 - 6.2.3. Load the **2D markers** with "Files" menu from:
"/data/Projects/year/project_name/C01/Etc/CAM1.marker".
 - 6.2.4. Load the **3D markers** with "Files" menu from:
"/data/Projects/year/project_name/C01/3D/CAM1.marker".
 - 6.2.5. **Align** the transformed markers with the 2D markers:
 - 6.2.5.1. This may be easier with the following settings: Show Trans. Marker, Errorbar, Marker Connections, Numbering, Quads.
 - 6.2.5.2. Try to **turn the model** such that the trans. markers are roughly aligned with the 2D markers
 - 6.2.5.3. Select "**Align**" in the 3D-GUI.
 - 6.2.5.4. In order to perform automatic alignment with nToPas, select "**Init**" and "**Iterate**". The error bar should turn green.
 - 6.2.5.5. Try to deselect and specify lens correction ("**l.c.**") or other parameters in order to achieve the best alignment which is possible.
 - 6.2.5.6. Select "**Get Pixel**" in order to map the 2D image on the 3D grid.

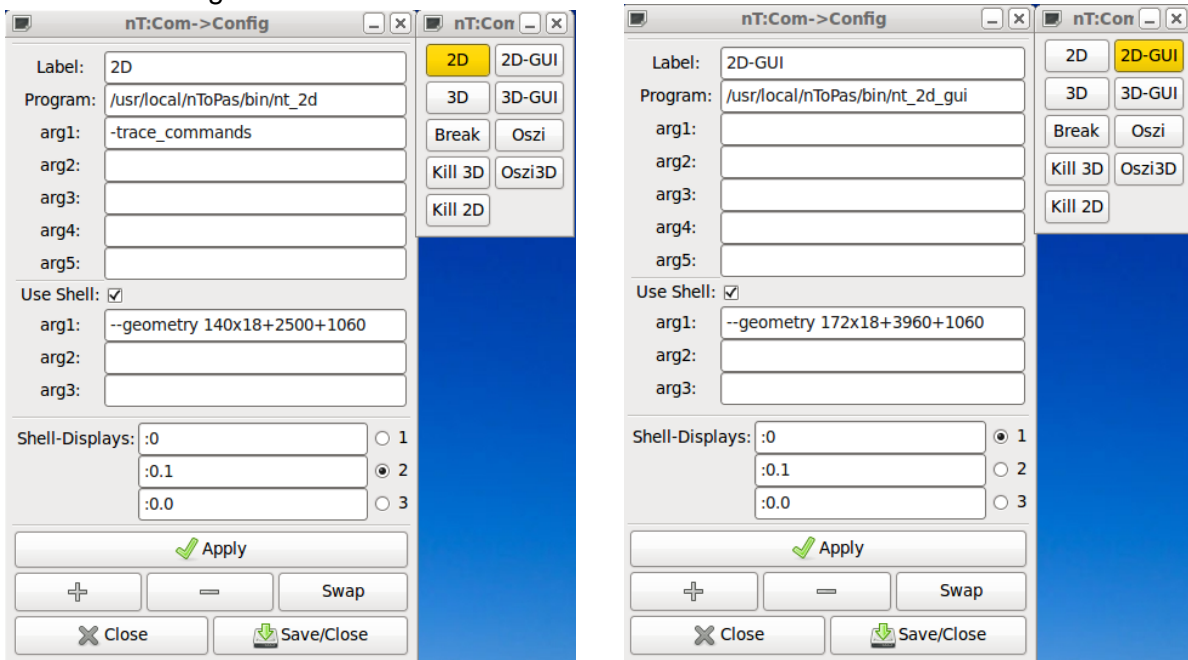
- 6.2.5.7. **Check the edges** of the 3D grid in order to see if the alignment was well done.
- 6.2.5.8. Repeat this process if necessary until good alignment is achieved.
- 6.2.5.9. In some cases the position of the 3D markers has to be adjusted by hand.
- 6.2.6. Save the point of view with "**Save POV**" for example as:
"/data/Projects/year/project_name/C01/3D/CAM1.pov". This is the starting point for the automatic 3D alignment with the nToPas perl scripts
- 7. Select "Batch" to open the Batch-window for automatic data processing (Picture 7)
 - 7.1. Select the measurement Point (C,S,P), which you want to process.
 - 7.2. Select the radio button on the right of the perl script. Then select the perl script, which you want to execute with the left mouse button.
 - 7.3. Usually the scripts have to be executed in order from top to bottom.
 - 7.4. If you want to edit a Perl-Script select it with the right mouse button.
- 8. **Installation of a new version of nToPas**
 - 8.1. Open a terminal (click on the black text window in the center of the Desktop header.)
 - 8.2. Change the path to "/usr/local" by typing: "**cd /usr/local**".
 - 8.3. Copy the installation file from the USB memory stick to the current directory: "**sudo cp /media/.../nToPas.400.08.tar.bz2 /usr/local**", where the number "400.08" specifies the version of nToPas.
 - 8.4. Remove old backup if necessary: "**sudo rm -r nToPas.old**".
 - 8.5. Make a backup from the last nToPas version: "**sudo cp -r nToPas nToPas.old**".
 - 8.6. Install new version of nToPas: "**sudo tar xvfj nToPas.400.08.tar.bz2**".
- 9. **Installation of a new perl modul for nToPas**
 - 9.1.1. Change the path to "/etc/perl/" by typing: "**cd /etc/perl**".
 - 9.1.2. Copy the old perl modul: "**sudo cp nToPas.pm nToPas.pm.old**".
 - 9.1.3. Copy the new perl modul from the USB memory to the current directory: "**sudo cp /media/.../nToPas.pm /etc/perl**"

Pictures for step-by-step manual for nToPas at ETW

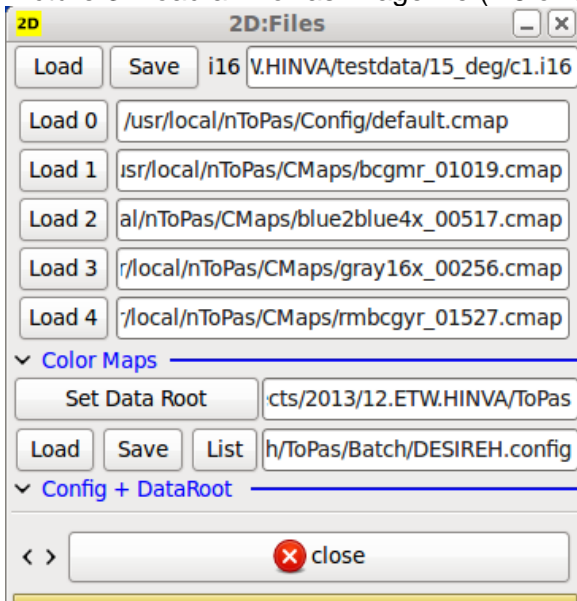
Picture 1: Start “2D” and “2D-GUI”



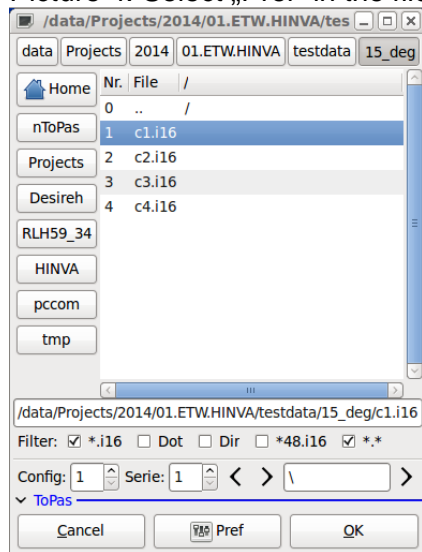
Picture 2: Change window location of “2D” and “2D-GUI” with the “Pref” button



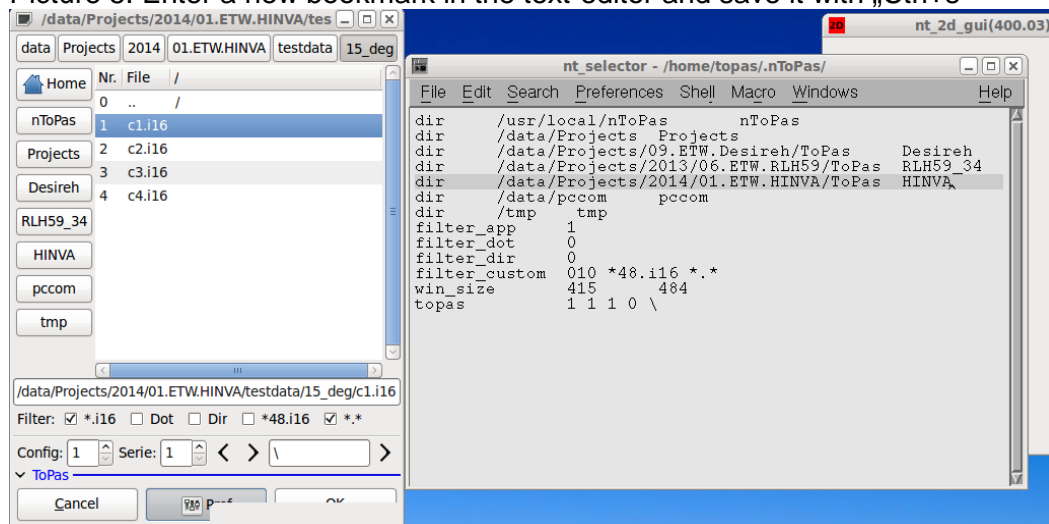
Picture 3: Load a nToPas image file (i16 or b16-format)



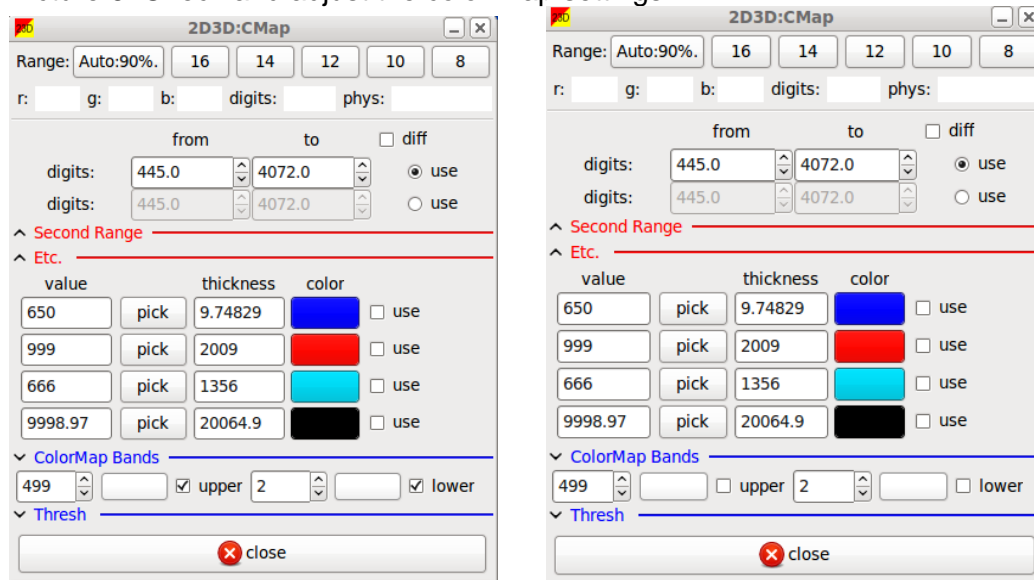
Picture 4: Select „Pref“ in the file selector



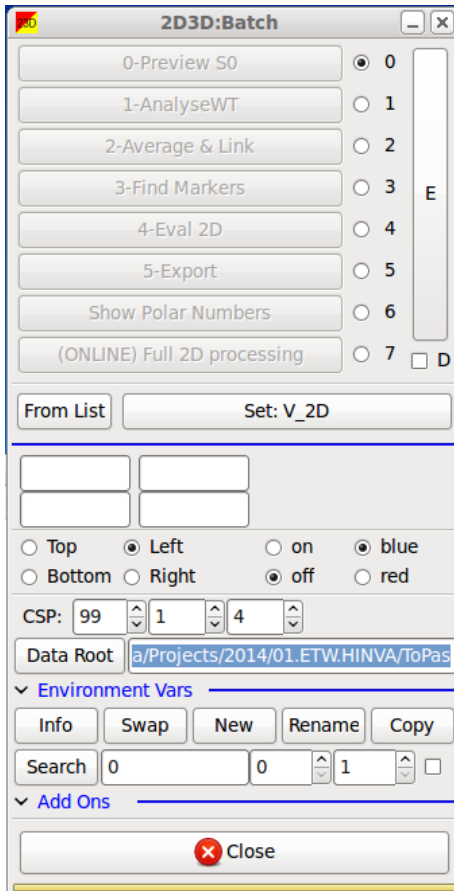
Picture 5: Enter a new bookmark in the text-editor and save it with „Ctrl+s“



Picture 6: Check and adjust the color map settings



Picture 7: The Batch-window for automatic data processing



The screenshot shows the '2D3D:Batch' window with the following settings:

- Process Steps:** A list of steps with radio buttons. Step 0 ('0-Preview S0') is selected. A vertical scrollbar is visible to the right of the list.
- From List:** A button labeled 'Set: V_2D'.
- Orientation and Color:**
 - Orientation: ☐ Top, ☒ Left, ☐ Bottom, ☐ Right.
 - Color: ☐ on, ☒ blue, ☐ off, ☐ red.
- CSP:** Three spinners showing values 99, 1, and 4.
- Data Root:** A text field containing the path 'a/Projects/2014/01.ETW.HINVA/ToPas'.
- Environment Vars:** A section with buttons 'Info', 'Swap', 'New', 'Rename', and 'Copy'.
- Search:** A section with a 'Search' label and three spinners showing values 0, 0, and 1.
- Add Ons:** A section with a 'Close' button (indicated by a red X icon).